Amendments to the Claims:

1.   (Currently Amended) An apparatus for smoothly playing a ~~pre-determined~~ predetermined sequence of streamed content segments ~~songs transmitted from a server over the Internet, the apparatus~~ comprising;

a processor;

a first memory that stores at least one control program ~~used by~~ usable by the processor to control ~~the playing~~ play of the predetermined sequence of ~~songs~~ content segments, the at least one control program including computer-readable instructions specifying a number of beginning portions of ~~songs~~ content segments to cache in advance and size of a pre-buffer cache; and

a second memory which is available to the at least one control program, wherein the ~~at least one control program causes the processor at least~~ apparatus is configured to:

~~as a song starts to play, start to download,~~ in response to initiation of play of a content segment, initiate downloading to the pre-buffer cache of a beginning portion of each of a number of ~~songs~~ content segments which are, in the ~~pre-determined~~ predetermined sequence, subsequent to the playing ~~song, to the pre-buffer cache~~ content segment, wherein the pre-buffer cache is an area of the second memory;

~~if playback is skipped to a target song for which the~~ in response to skipping to a target content segment of the predetermined sequence of content segments whose beginning portion has been downloaded to the pre-buffer cache, ~~start to~~ initiate play of the downloaded beginning portion of the ~~target song~~ target content segment if less than a preallocated quantity of content segments were previously streamed, during a subscription period, in association with a subscriber; and

while playing the downloaded beginning portion of the target ~~song, start to download~~ content segment, initiate downloading of the rest of the ~~target song~~ target content segment.

2.   (Currently Amended) The apparatus of claim 1, wherein the beginning portion of the ~~target song~~ target content segment is approximately the data of the first ten seconds of the ~~target song~~ target content segment.

3.   (Currently Amended) The apparatus of claim 1, wherein the number of beginning portions of ~~songs~~ content segments to cache in advance is five.

4.   (Currently Amended) The apparatus of claim 1, wherein the number of beginning portions of ~~songs~~ content segments to cache in advance is all ~~songs~~ content segments in the ~~pre-determined~~ predetermined sequence of ~~songs~~ content segments that are subsequent to the ~~playing song~~ playing content segment.

5.   (Previously Presented) The apparatus of claim 1, wherein the pre-buffer cache follows a first-in first-out algorithm and allows writing while reading.

6.   (Currently Amended) A method for smoothly playing a ~~pre-determined~~ predetermined sequence of streamed content segments ~~songs transmitted from a remote server to a local playback device over the Internet~~, comprising:
    ~~as a song starts~~ in response to initiation of play of a content segment on ~~the local~~ a local playback device, downloading, consecutively, a beginning portion of each of a number of ~~songs~~ content segments which are, in the ~~pre-determined~~ predetermined sequence, subsequent to the ~~playing song~~ playing content segment, wherein the initiation of play of the content segment is based on whether less than a preallocated quantity of content segments were previously streamed, during a subscription period, in association with a subscriber; and
    pre-caching the downloaded beginning portions to a pre-buffer cache of a memory of the local playback device, wherein the number of beginning portions of ~~songs~~ content segments to pre-cache in advance and size of the pre-buffer cache are specified by a function call.

7.   (Currently Amended) The method of claim 6, further comprising:

~~if playback is skipped from a playing song to a target song,~~ in response to skipping to a target content segment of the predetermined sequence of content segments, checking whether the beginning portion or the ~~target song~~ target content segment is in the pre-buffer cache; and

if the beginning portion of the ~~target song~~ target content segment is in the pre-buffer cache, playing the beginning portion of the ~~target song~~ target content segment from the pre-buffer cache, downloading at least a portion of ~~the song~~ the content segment which is not in the pre-buffer cache, and deleting beginning portions of any content segments ~~songs~~ prior to the ~~target song~~ target content segment in the ~~predetermined~~ predetermined sequence from the pre-buffer cache.

8.   (Currently Amended) The method of claim 7, further comprising:

if the beginning portion of the ~~target song~~ target content segment is in the pre-buffer cache, downloading, consecutively, a beginning portion of each of a number of ~~songs~~ content segments which are in the ~~pre-determined~~ predetermined sequence, subsequent to the ~~target song~~ target content segment, wherein if beginning portions of the one or more ~~songs~~ content segments subsequent to the ~~target song~~ target content segment are already in the pre-buffer cache, skipping the downloading the beginning portions of the one or more ~~songs~~ content segments already having beginning portions in the pre-buffer cache and downloading the beginning portions of the subsequent ~~songs, consecutively, to make up the number of~~ content segments such that beginning portions of ~~songs~~ each of the number of content segments to pre-cache in advance are downloaded to the pre-buffer cache.

9.   (Currently Amended) The method of claim 8, further comprising:

if no skip command is received while the ~~target song~~ target content segment is playing, as the playing of the ~~target song~~ target content segment ends, playing ~~the song~~ the content segment immediately subsequent to the ~~target song~~ target content segment; and

if a skip command is received while the ~~target song~~ target content segment is playing, checking whether the beginning portion of ~~the song~~ the content segment immediately subsequent to the ~~target song~~ target content segment is in the pre-buffer cache.

10.   (Currently Amended) The method of claim 7, wherein if the beginning portion of the ~~target song~~ target content segment is not in the pre-buffer cache, the method further comprises:

sending a request to stop transmitting the ~~playing song~~ playing content segment and to start transmitting the ~~target song~~ target content segment and at least substantially simultaneously:

deleting the beginning portion of ~~any song~~ any content segment which is prior to the ~~target song~~ target content segment in the ~~pre-determined~~ predetermined sequence of ~~songs~~ content segments from the pre-buffer cache;

downloading at least a remaining portion of the ~~target song~~ target content segment; and

begin playing the ~~target song as~~ target content segment when a sufficient portion of the ~~target song~~ target content segment has been downloaded.

11.   (Currently Amended) The method of claim 10, further comprising:

if playback is skipped from the ~~target song~~ target content segment to another ~~target song~~ target content segment, checking whether the beginning portion of the other ~~target song~~ target content segment is in the pre-buffer cache; and

if playback is not skipped from the ~~target song~~ target content segment, playing the beginning portion of ~~the song~~ content segment subsequent to the ~~target song~~ target content segment after the end of the ~~target song~~ target content segment is played and downloading at least a portion of the ~~target song~~ target content segment which is not in the pre-buffer cache, wherein if beginning portions of the one or more ~~songs~~ content segments subsequent to ~~songs~~ content segments in the ~~pre-determined~~ predetermined sequence of ~~songs~~ content segments are already in the pre-buffer cache, skipping the

downloading of the beginning portions of the one or more ~~songs~~ content segments already having beginning portions in the pre-buffer cache and downloading the beginning portions of the subsequent ~~songs, consecutively, to make up the number of~~ content segments such that beginning portions of ~~songs~~ each of the number of content segments to pre-cache in advance are downloaded to the pre-buffer cache.

12.   (Currently Amended) The method of claim 6, wherein the beginning portion of the ~~target song~~ target content segment is approximately the data of the first ten seconds of the ~~target song~~ target content segment.

13.   (Currently Amended) The method of claim 6, wherein the number of beginning portions of ~~songs~~ content segments to cache in advance is five.

14.   (Currently Amended) The method of claim 6, wherein the number of beginning portions of ~~songs~~ content segments to cache in advance is all ~~songs~~ content segments in the ~~pre-determined~~ predetermined sequence of ~~sense~~ content segments that are subsequent to the ~~playing song~~ playing content segment.

15.   (Previously Presented) The method of claim 6, wherein the pre-buffer cache follows a first-in first-out algorithm and allows writing while reading.

16.   (Currently Amended) A ~~program~~ computer-readable storage medium ~~readable by a computer, tangibly embodying a program of~~ having instructions ~~executable by the computer to perform a method for smoothly playing~~ stored thereon that, if executed by a computing device, cause the computing device to perform operations which play a ~~pre-determined~~ predetermined sequence of ~~songs transmitted from a remote server to the computer over the Internet~~ streamed content segments, the operations comprising:

as a ~~song starts~~ in response to initiation of play of a content segment on the ~~computer~~ computing device, downloading a beginning portion of each of a number of

~~songs~~ content segments which are, in the ~~pre-determined~~ predetermined sequence, subsequent to the ~~playing song~~ playing content segment, wherein the initiation of play of the content segment is based on whether less than a preallocated quantity of content segments were previously streamed, during a subscription period, in association with a subscriber; and

pre-caching the downloaded beginning portions to a pre-buffer cache of a memory of the ~~local playback~~ computing device, wherein the number of beginning portions of ~~songs~~ content segments to pre-cache in advance and size of the pre-buffer cache are specified by a function call.

17.   (Currently Amended) The ~~program~~ computer-readable storage medium of claim 16, wherein the operations further ~~comprising~~ comprise:

~~If playback is skipped from a playing song to a target song,~~ in response to skipping to a target content segment of the predetermined sequence of content segments, checking whether the beginning portion of the ~~target song~~ target content segment is in the pre-buffer cache; and

If the beginning portion of the ~~target song~~ target content segment is in the pre-buffer cache, playing the beginning portion of the ~~target song~~ target content segment from the pre-buffer cache, downloading at least a portion of the ~~target song~~ target content segment which is not in the pre-buffer cache, and deleting beginning portions of any ~~songs~~ content segments prior to the ~~target song~~ target content segment in the ~~pre-determined~~ predetermined sequence from the pre-buffer cache.

18.   (Currently Amended) The ~~program~~ computer-readable storage medium of claim 17, wherein the operations further ~~comprising~~ comprise:

if the beginning portion of the ~~target song~~ target content segment is in the pre-buffer cache, downloading a beginning portion each a numbers of ~~songs~~ content segments which are, in the ~~pre-determined~~ predetermined sequence, subsequent to the ~~target song~~ target content segment, wherein if beginning portions of the one or more ~~songs~~ content segments subsequent to the ~~target song~~ target content segment are

already in the pre-buffer cache, skipping the downloading of the beginning portions of the one or more ~~songs~~ content segments already having beginning portions in the pre-buffer cache and downloading the beginning portions of the subsequent ~~songs,~~ ~~consecutively, to make up the number of~~ content segments such that beginning portions of ~~songs~~ each of the number of content segments to pre-cache in advance are downloaded to the pre-buffer cache.

19.   (Currently Amended) The ~~program~~ computer-readable storage medium of claim 18, wherein the operations further ~~comprising~~ comprise:

if no skip command is received while the ~~target song~~ target content segment is playing, as the playing of the ~~target song~~ target content segment ends, playing ~~the song~~ the content segment immediately subsequent to the ~~target song~~ target content segment; and

if a skip command is received while the ~~target song~~ target content segment is playing, checking whether the beginning portion of ~~the song~~ the content segment immediately subsequent the ~~target song~~ target content segment is in the pre-buffer cache.

20.   (Currently Amended) The ~~program~~ computer-readable storage medium of claim 17, wherein if the beginning portion of the ~~target song~~ target content segment is not in the pre-buffer cache, the ~~method~~ operations further ~~comprises~~ comprise:

sending a request to stop transmitting the ~~playing song~~ playing content segment and to start transmitting the ~~target song~~ target content segment, and at least substantially simultaneously:

deleting the beginning portion of ~~any song~~ any content segment which is prior to the ~~target song~~ target content segment in the ~~pre-determined~~ predetermined sequence of ~~songs~~ content segments from the pre-buffer ~~cache, and~~ cache;

downloading at least a remaining portion of the ~~target song~~ target content segment; and

begin playing the ~~target song as~~ target content segment after a sufficient portion of the ~~target song~~ target content segment has been downloaded.

21.   (Currently Amended) The ~~program~~ computer-readable storage medium of claim 20, further ~~comprising~~ comprise:

If playback is skipped from the ~~target song~~ target content segment to another ~~target song~~ target content segment, checking whether the beginning portion of the other ~~target song~~ target content segment is in the pre-buffer cache; and

if playback is not skipped from the ~~target song~~ target content segment, playing the beginning portion of ~~the song~~ the content segment subsequent to the ~~target song~~ target content segment after the end of the ~~target song~~ target content segment is played, downloading at least a portion of the ~~target song~~ target content segment which is not in the pre-buffer cache, wherein if beginning portions of the one or more ~~songs~~ content segments subsequent to ~~songs~~ content segments in the ~~pre-determined~~ predetermined sequence are already in the pre-buffer cache, skipping the downloading of the beginning portions of one or more ~~songs~~ content segments already having beginning portions in the pre-buffer cache and downloading the beginning portions of the subsequent ~~songs to make up the number of~~ content segments such that beginning portions of ~~songs~~ each of the number of content segments to cache in advance are downloaded to the pre-buffer cache.

22.   (Currently Amended) The ~~program~~ computer-readable storage medium of claim 16, wherein the beginning portion of the ~~target song~~ target content segment is approximately the data of the first ten seconds of the ~~target song~~ target content segment.

23.   (Currently Amended) The ~~program~~ computer-readable storage medium of claim 16, wherein the number of beginning portions of ~~songs~~ content segments to cache in advance is five.

24.   (Currently Amended) The ~~program~~ computer-readable storage medium of claim 16, wherein the number of beginning portions of ~~songs~~ content segments to cache in advance is all ~~songs~~ content segments in the ~~pre-determined~~ predetermined sequence of ~~songs~~ content segments that are subsequent to the ~~playing song~~ playing content segment.

25.   (Currently Amended) The ~~program~~ computer-readable storage medium of claim 16, wherein the pre-buffer cache follows a first-in first-out algorithm and allows writing while reading.

26.   (Currently Amended) An apparatus for smoothly playing a ~~pre-determined~~ predetermined sequence of ~~songs transmitted from a server over the Internet~~ streamed content segments, comprising~~;~~:

means for controlling the playback of a ~~pre-determined~~ predetermined sequence of ~~songs~~ content segments based on whether less than a preallocated quantity of content segments were previously streamed, during a subscription period, in association with a subscriber;

means for pre-downloading a beginning portion of a number of ~~songs~~ content segments from the ~~pre-determined~~ predetermined sequence of ~~songs~~ content segments; and

means for caching the pre-downloaded beginning portions, wherein the number of beginning portions to be pre-downloaded is configurable via a function call.